AMENDMENTS TO THE CLAIMS

1. (Previously Presented) A computer-implemented method for enabling the registration of dynamically generated code and corresponding unwind information, said method comprising:

creating a module which includes data related to said dynamically generated code and said corresponding unwind information, said dynamically generated code comprising instrumented code for a function;

providing an application program interface which allows said data to be registered such that dynamic registration of said dynamically generated code and said corresponding unwind information is enabled;

patching an entry point of said function to an entry point of said instrumented code; and

setting an instruction pointer to the beginning of an application program interface invocation code sequence that precedes said entry point of said instrumented code.

2. (Original) The computer-implemented method for enabling the registration of dynamically generated code and corresponding unwind information as recited in Claim 1 wherein said module stores said data related to said dynamically generated code and said corresponding unwind information in a centralized location.

3. (Canceled).

4. (Original) The computer-implemented method for enabling the registration of dynamically generated code and corresponding unwind information as recited in Claim 1 wherein said application program interface allows said data to be registered by a dynamic loader.

5. (Previously Presented) A computer-readable medium embodying instructions that cause a computer to perform a method for enabling the registration of dynamically generated code and corresponding unwind information, said method comprising:

creating a module which includes data related to said dynamically generated code and said corresponding unwind information, said dynamically generated code comprising instrumented code for a function;

providing an application program interface which allows said data to be registered such that dynamic registration of said dynamically generated code and said corresponding unwind information is enabled;

patching an entry point of said function to an entry point of said instrumented code; and

setting an instruction pointer to the beginning of an application program interface invocation code sequence that precedes said entry point of said instrumented code.

6. (Original) The computer-readable medium of Claim 5 wherein said module stores said data related to said dynamically generated code and said corresponding unwind information in a centralized location.

7. (Canceled).

8. (Original) The computer-readable medium of Claim 5 wherein said application program interface allows said data to be registered by a dynamic loader.

9. (Previously Presented) An apparatus for enabling the registration of dynamically generated code and corresponding unwind information, the apparatus comprising:

means for creating a module which includes data related to said dynamically generated code and said corresponding unwind information, said dynamically generated code comprising instrumented code for a function;

means for providing an application program interface which allows said data to be registered such that dynamic registration of said dynamically generated code and said corresponding unwind information is enabled;

means for patching an entry point of said function to an entry point of said instrumented code; and

means for setting an instruction pointer to the beginning of an application program interface invocation code sequence that precedes said entry point of said instrumented code.

10. (Original) The apparatus of Claim 9 wherein said module stores said data related to said dynamically generated code and said corresponding unwind information in a centralized location.

11. (Canceled).

12. (Original) The apparatus of Claim 9 wherein said application program interface allows said data to be registered by a dynamic loader.

13. (Currently Amended) A computer-implemented method for registering dynamically generated code for a function and for registering corresponding unwind information, said method comprising:

creating a module which includes data related to said dynamically generated code and said corresponding unwind information;

providing an application program interface which allows said data to be registered such that dynamic registration of said dynamically generated code and said corresponding unwind information is enabled;

coupling an application program interface invocation code sequence to said dynamically generated code such that upon execution of said dynamically generated code [[,]] ~~wherein~~ said application program interface invocation code sequence instructs said application program interface to facilitate registration of said data;

patching an entry point of said function to an entry point associated with said dynamically generated code; and

setting an instruction pointer to the beginning of said application program interface invocation code sequence.

14. (Original) The computer-implemented method for registering dynamically generated code and corresponding unwind information as recited in Claim 13 wherein said module stores said data related to said dynamically generated code and said corresponding unwind information in a centralized location.

15. (Original) The computer-implemented method for registering dynamically generated code and corresponding unwind information as recited in Claim 13 wherein said dynamically generated code is comprised of instrumented code.

16. (Original) The computer-implemented method for registering dynamically generated code and corresponding unwind information as recited in Claim 13 wherein said application program interface allows said data to be registered by a dynamic loader.

17. (Currently Amended) The computer-implemented method for registering dynamically generated code and corresponding unwind information as recited in Claim 13 wherein space for storing said application program interface invocation code sequence is reused for a second application program interface invocation code sequence for ~~utilized by a~~ second dynamically generated code for a second function.

18. (Original) The computer-implemented method for registering dynamically generated code and corresponding unwind information as recited in Claim 13 further comprising:

generating a second application program interface invocation code sequence for coupling to second dynamically generated code and corresponding unwind information such that upon execution of said second dynamically generated code, said second application program interface invocation code sequence instructs said application program interface to facilitate registration of data related to said second dynamically generated code and said corresponding unwind information.

19. (Original) The computer-implemented method for registering dynamically generated code and corresponding unwind information as recited in Claim 13 further comprising:

preventing registration of said module for a function called directly or indirectly via said application program interface.

20. (Previously Presented) The computer-implemented method for registering dynamically generated code and corresponding unwind information as recited in Claim 13 further comprising:

saving and restoring register values upon entry and exit of said application program interface invocation code sequence.

21. (Currently Amended) A computer-readable medium embodying instructions that cause a computer to perform a method for registering dynamically generated code for a function and for registering corresponding unwind information, said method comprising:

creating a module which includes data related to said dynamically generated code and said corresponding unwind information;

providing an application program interface which allows said data to be registered such that dynamic registration of said dynamically generated code and said corresponding unwind information is enabled;

coupling an application program interface invocation code sequence to said dynamically generated code such that upon execution of said dynamically generated code [[,]] ~~wherein~~ said application program interface invocation code sequence instructs said application program interface to facilitate registration of said data;

patching an entry point of said function to an entry point associated with said dynamically generated code; and

setting an instruction pointer to the beginning of said application program interface invocation code sequence.

22. (Original) The computer-readable medium of Claim 21 wherein said module stores said data related to said dynamically generated code and said corresponding unwind information in a centralized location.

23. (Original) The computer-readable medium of Claim 21 wherein said dynamically generated code is comprised of instrumented code.

24. (Original) The computer-readable medium of Claim 21 wherein said application program interface allows said data to be registered by a dynamic loader.

25. (Currently Amended) The computer-readable medium of Claim 21 wherein space for storing said application program interface invocation code sequence is reused for a second application program interface invocation code sequence for ~~utilized by a~~ second dynamically generated code for a second function.

26. (Previously Presented) The computer-readable medium of Claim 21 wherein said method further comprises:

generating a second application program interface invocation code sequence for coupling to second dynamically generated code and corresponding unwind information such that upon execution of said second dynamically generated code, said second application program interface invocation code sequence instructs said application program interface to facilitate registration of data related to said second dynamically generated code and said corresponding unwind information.

27. (Previously Presented) The computer-readable medium of Claim 21 wherein said method further comprises:

preventing registration of said module for a function called directly or indirectly via said application program interface.

28. (Previously Presented) The computer-readable medium of Claim 21 wherein said method further comprises:

saving and restoring register values upon entry and exit of said application program interface invocation code sequence.

29. (Currently Amended) An apparatus for registering dynamically generated code for a function and for registering corresponding unwind information, said apparatus comprising:

means for creating a module which includes data related to said dynamically generated code and said corresponding unwind information;

means for providing an application program interface which allows said data to be registered such that dynamic registration of said dynamically generated code and said corresponding unwind information is enabled;

means for coupling an application program interface invocation code sequence to said dynamically generated code such that upon execution of said dynamically generated code [[,]] ~~wherein~~ said application program interface invocation code sequence instructs said application program interface to facilitate registration of said data;

means for patching an entry point of said function to an entry point associated with said dynamically generated code; and

means for setting an instruction pointer to the beginning of said application program interface invocation code sequence.

30. (Original) The apparatus of Claim 29 wherein said module stores said data related to said dynamically generated code and said corresponding unwind information in a centralized location.

31. (Original) The apparatus of Claim 29 wherein said dynamically generated code is comprised of instrumented code.

32. (Original) The apparatus of Claim 29 wherein said application program interface allows said data to be registered by a dynamic loader.

33. (Currently Amended) The apparatus of Claim 29 wherein space for storing said application program interface invocation code sequence is reused for a second application program interface invocation code sequence for ~~utilized by a~~ second dynamically generated code for a second function.

34. (Original) The apparatus of Claim 29 further comprising:

means for generating a second application program interface invocation code sequence for coupling to second dynamically generated code and corresponding unwind information such that upon execution of said second dynamically generated code, said second application program interface invocation code sequence instructs said application program interface to facilitate registration of data related to said second dynamically generated code and said corresponding unwind information.

35. (Original) The apparatus of Claim 29 further comprising:

means for preventing registration of said module for a function called directly or indirectly via said application program interface.

36. (Previously Presented) The apparatus of Claim 29 further comprising:

means for saving and restoring register values upon entry and exit of said application program interface invocation code sequence.